프랙탈 압축을 위한 레인지 블록간의 유사성 분석

김 영 봉[†]

요 약

프랙탈 영상 압축은 영상의 일부 영역이 같은 영상의 다른 영역과 거의 유사한 모양을 하고 있다는 자기유사성에 기초하고 있다. 이 압축 방법은 높은 압축률과 빠른 복원력을 제공하지만 매우 긴 압축 시간을 갖는 단점이 있다. 압축 시간을 단축하기 위해, 가장 많은 시간이 소요되는 레인지 블록과 도메인 블록간의 비교 탐색 과정을 줄이는 연구가 꾸준히 이루어져 왔다. 이 연구들은 도메인 블록과 레인지 블록의 유사성에 기초한 탐색 시간의 단축 연구가 주를 이루고 있고, 레인지 블록들간에 존재하는 유사성을 적절히 이용하지 못하고 있다. 그리하여 본 연구에서는 레인지 블록간의 유사성을 이용한 압축 알고리즘의 개발을 위해 레인지 영역간의 유사성에 대한 분석을 수행할 것이다. 레인지 블록의 유사성을 결정하는 유사성 척도와 문턱치에 따른 유사성 정도를 비교 및 분석할 것이다. 이 제안 방법은 다른 프랙탈 영상 압축 기법의 사전 작업으로 활용되어 더욱 큰 효과를 볼 것으로 기대된다.

An Analysis on Range Block Coherences for Fractal Compression

Kim, Young-Bong[†]

ABSTRACT

The fractal image compression is based on the self-similarity that some area in an image exhibits a very similar shape with other areas. This compression technique has very long encoding time although it has high compression ratio and fast decompression. To cut-off the encoding time, most researches have restricted the search of domain blocks for a range block. These researches have been mainly focused on the coherence between a domain block and a range block, while they have not utilized the coherence among range blocks well. Therefore, we give an analysis on the coherence among range blocks in order to develope an efficient fractal image compression algorithm. We analysis the range blocks according to not only measures for defining the range block coherence but also threshold of each measure. If these results are joined in a prior work of other fractal compression algorithms, it will give a great effectiveness in encoding time.

1. 서 론

프랙탈 압축 알고리즘은 Jacquin이 단색 디지털 영상에 기초하여 제시한 이후 활발히 연구가 이루어져 오고 있다. 이 압축 기법은 일반 영상 내의 일부 영역이 같은 영상의 다른 영역과 매우 유사한 모양을 하고 있다는 프랙탈의 자기유사성(self similarity)에 기초한 것이다. 밝기 값의 변화가 유사한 두 영역인 레인지 블록(range block)과 도메인 블록(domain block) 간의 탐색과 두 블록간의 아핀 변환(affine transformation)을 찾는 작업이 프랙탈 압축의 주요 과제이다[1-4]. 프랙탈 압축 결과인 아핀 변환을 임의의 영상에 반복적으로 적용시키면 원 영상으로 수렴하는 특징을 가지기 때문에 아핀 변환의 계수 값만 저장하게되어 높은 압축율을 얻을 수 있다. 그러나 이 방법은 압축에 소요되는 시간이 매우 길다는 단점을 가지고 있다.

프랙탈 영상 압축 시간을 단축하기 위한 방법으로

본 논문은 1999년도 부경대학교 기성회 연구비 지원을 받았음

† 부경대학교 컴퓨터멀티미디어공학부

각 레인지 블록에 대한 도메인 블록의 탐색 횟수를 줄이는 많은 연구들이 이루어져 왔다. 이 연구들은 크게 두 가지 접근 방법으로 나누어 볼 수 있다. 첫 번째 방법은 레인지 블록과 유사한 도메인 블록을 찾을 때 모든 도메인 블록을 검사하지 않고 두 블록간의 차이를 나타내는 측정값이 주어진 기준에 부합하는 최초의 블록을 선택하는 방법이다[5,6]. 이 방법에서는 도메인 블록을 찾는 순서에 따라 수행 결과가 다르게 되며, 이에 따른 영상의 품질에도 차이를 보이게 된다. Barthel 등은 레인지 블록을 내포하는 도메인 블록들 중에서 유사 블록이 발견될 확률이 높다는 사실에 기초한 도메인 블록의 탐색 순서를 결정한 방법을 제시하고 있다[5].

두 번째 방법은 레인지 블록과 도메인 블록의 이미지 특성을 이용한 것으로 각 레인지 블록에 대해 유사한 특징을 갖고있는 도메인 블록들만을 탐색하는 방법이다. 이 방법에서는 DCT등의 영상 변환 기법을 적용한 후 각 블록 영상의 특징을 나타낼 수 있는 몇 개의 데이터를 사용하여 도메인 블록과 레인지 블록간의 유사성을 측정하게 된다. 임의의 레인지 블록에 대한 영상 특징 값과 유사한 영상 특징 값을 보여주는 도메인 블록들과만 비교 검사하는 방법이다[7-10]. 즉, 모든 도메인 블록을 영상의 특징에 따라 몇 개의 집합으로 분류하는 작업이 선행된 뒤, 각 레인지 블록은 유사한 특징을 갖는 도메인 블록의 집합에 속한 블록과만 비교가 이루어진다.

프랙탈 압축 알고리즘들에서 대부분의 수행시간은 도메인 영역의 탐색에 집중되어 있다. 따라서 레인지 블록 각각에 대해 많은 수의 도메인 블록들을 매번 탐색해야 하므로 부호화 시간이 기하급수적으로 증가하게 된다. 만일 어떤 두 개의 레인지 블록이 유사한 밝기 패턴을 가진다면 두 블록에 대한 도메인 영역의 탐색 결과는 거의 같을 것이다. 만약 서로 다른 도메인 블록을 선택했다 하더라도 같은 도메인 블록을 사용한 결과와 비교할 때 큰 차이를 보이지 않을 것이다. 다시 말해, 두 개의 레인지 블록에 같은 도메인 블록을 적용하여 각 레인지 블록에 대한 프랙탈 코드를 계산해도 결과에는 큰 영향을 미치지 않을 것이다.

그리하여, 본 연구에서는 도메인 블록의 탐색이 필요한 레인지 블록의 수를 줄이기 위한 기초 연구로써 일반 영상의 레인지 블록간의 유사성에 대한 분석을 수행할 것이다. 즉, 레인지 블록간의 유사성을 판단하는 다양한 척도의 개발 및 그에 따른 레인지 블록간의 그룹화 결과에 대한 비교와 분석 작업을 수행할 것이다. 본 논문의 구성은 2장에서 일반적인 프랙탈 영상 압축 알고리즘을 소개하고 3장에서는 본 논문에서 구현한 다양한 척도에 따른 레인지 영역의 유사성을 찾는 방법을 기술할 것이다. 그리고 4장에서 여러 영상에 대해 레인지 유사성에 의한 레인지 블록의 그룹화 결과를 통해 레인지 블록의 유사성을 이용한 방법의 효율성을 보여주고 5장에서 결론을 맺는다.

2. 프랙탈 영상 부호화

2.1 프랙탈 압축 시스템

프랙탈 압축은 프랙탈의 기본 특징중 하나인 자기유사성을 이용한 것이다. 즉, 임의의 2차원 영상을 각 픽셀의 밝기 값을 높이로 고려한 3차원 입체에서 특정 지역과 유사한 모습을 하고 있는 다른 지역을 찾는 작업으로 이루어진다. 프랙탈은 초기 영상에 관계없이 두 유사 지역간의 변환 규칙을 반복적으로 적용하면 최종적인 영상이 결정된다는 특성을 가지고 있다. 따라서 임의의 2차원 영상에서 출발해 원 영상을 복원할 수 있는 각각의 레인지 블록에 대한 변환규칙인 IFS(iterated function system)를 찾는 것이 프랙탈 압축 알고리즘의 핵심 과제이다. IFS란 두 영상간의 변환을 회전(rotation), 신축(scaling), 반사(reflection) 및 이동(translation)의 조합으로 나타낸 것으로 식(1)에 기술된 행렬로 표현된다.

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & 0 & e \\ c & d & 0 & f \\ 0 & 0 & s & o \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{1}$$

식(1)에서 x, y 는 영상 평면상의 주어진 픽셀의 위치를 가리키고, z는 해당 픽셀의 밝기 값을 나타낸다. 그리고 아핀 변환 행렬에서 a, b, c, d, e, f는 기하 변환을 나타내고, s는 두 블록간의 contrast scaling, o는 두 블록간의 평균 화소 밝기 값의 차이인 offset을 나타낸다.

프랙탈 압축 영상의 복원은 획득된 IFS를 임의의 영상에 반복적으로 적용하여 이루어지므로 IFS는 수렴하는 특징을 가져야한다. 즉, 영상을 구성하는

특정 영역에 대한 IFS인 아핀 변환은 아래 식을 만족하는 축소 변환(contractive transform)이 되어야 한다.

$$d(W(A),\ W(B)) \le r \cdot d(A,\ B) \qquad \forall A, B \in U,\quad 0 \le r < 1 \tag{2}$$

여기서 U는 완전 측도 공간(complete metric space)이며, d는 측도(metric)이고, r은 0과 1사이의 값으로 변환 W의 수축도(contractivity)를 나타낸다. 여기서 r 의 크기가 수렴의 정도를 나타내는 것으로 r 값이 1에 가까울수록 두 영상 A와 B가 천천히 같은 영상으로 바뀌게 된다.

2.2 레인지 블록과 도메인 블록

일반적인 프랙탈 압축 알고리즘은 전체 영상을 1개의 대상으로 한 IFS의 도출은 계산 량이 너무 방대할 뿐만 아니라 영상의 표현에도 부적합하기 때문에 영상을 여러 블록으로 나눈 후 각 영역 별로 아핀 변환을 찾는 방법을 사용한다. 이를 위해 영상을 같은 크기의 겹침이 없는 m×m 크기의 정사각형 영역인 레인지 블록으로 분할한다. 그리고 도메인 블록은 일반적으로 레인지 블록의 2배인 2m×2m 크기로 겹침을 허용하면서 분할한 사각형 영역으로 정의한다. 따라서 256×256 영상에서 m=8인 경우에 총 레인지 블록은 (256/8) × (256/8) = 1024개가 만들어지고, 도메인 블록은 (256-15)×(256-15)=58081개가 생성된다. 레인지 블록의 집합을 레인지 영역, 도메인 블록의 집합을 도메인 영역이라 부른다. 일반적으로 m 값의 크기는 8을 가장 많이 사용한다.

2.3 도메인 블록의 탐색

영상의 프랙탈 압축은 영상을 레인지 블록으로 분리한 후 각 레인지 블록에 대한 IFS를 구하게 된다. 이때 각 레인지 블록에 대해 유사한 밝기 변화를 보이는 도메인 블록을 찾는 작업이 필요하다. 이때 도메인 블록은 식(1)에 주어진 기하 변환이 가해진 상태로 레인지 블록과 비교된다. 그러나, 도메인 블록은 2차원 비트맵 영상으로 다양한 기하 변환을 적용시키기가 매우 어렵기 때문에 식(1)에서 영상의 기하 변환 계수들인 a, b, c, d에 제한을 가한 표1의 8가지 등장변환(isometry)을 주로 사용하게 된다.

표1의 8가지 등장변환이 도메인 블록에 적용되어 레인지 블록과 가장 유사한 블록과 변환을 찾게 된다. 식(1)의 e와 f는 도메인 블록의 위치를 나타내고, 두 블록간의 밝기 값을 나타내는 면의 contrast scaling 과 offset 값인 s와 o는 식(3)으로 구한다.

$$s = \left[n\sum_{i=0}^{n} d_i r_i - \sum_{i=0}^{n} d_i \sum_{i=0}^{n} r_i\right] / \left[n\sum_{i=0}^{n} d_i^2 - \left(\sum_{i=0}^{n} d_i\right)^2\right]$$
$$o = \frac{1}{n}\left[\sum_{i=0}^{n} r_i - s\sum_{i=0}^{n} d_i\right] \tag{3}$$

여기서 r_i와 d_i는 레인지 블록과 변환된 도메인 블록을 구성하는 각 픽셀 값을 나타낸다.

레인지 블록과 변환이 적용된 도메인 블록간의 유사성 측정을 위해 일반적으로 MSE(mean square error) 값을 활용한다. 두 블록간의 MSE 값은 식(4)와 같이 레인지 블록 R을 구성하는 각 픽셀의 밝기값 $r_1, r_2, \ldots, r_n$과 변환된 도메인 블록 D를 1/2로 축소한 축소 변환 도메인 블록 D'을 구성하는 각 픽셀의 밝기값 $d_1, d_2, \ldots, d_n$간의 차의 제곱 평균이 된다.

$$MSE = \frac{SSE}{n} = \frac{\sum_{i=1}^{n}(s \cdot d_i + o - r_i)^2}{n} \tag{4}$$

전체 도메인 영역의 탐색에서 MSE 값을 최소로 하는 도메인 블록과 변환 행렬이 결정되면 식(1)의 변환 계수들 (a, b, c, d, e, f, s, o)이 레인지 블록에 대한 프랙탈 코드로 사용된다.

3. 레인지 블록간의 유사도 측정

여러 레인지 블록에 같은 도메인 블록을 대응시키기 위해서는 레인지 블록간에 유사성이 존재 해야한

표 1. 8가지 등장변환

변환	0	1	2	3	4	5	6	7
행렬	$\begin{pmatrix}1 & 0\\0 & 1\end{pmatrix}$	$\begin{pmatrix}-1 & 0\\0 & 1\end{pmatrix}$	$\begin{pmatrix}1 & 0\\0 & -1\end{pmatrix}$	$\begin{pmatrix}-1 & 0\\0 & -1\end{pmatrix}$	$\begin{pmatrix}0 & 1\\1 & 0\end{pmatrix}$	$\begin{pmatrix}0 & 1\\-1 & 0\end{pmatrix}$	$\begin{pmatrix}0 & -1\\1 & 0\end{pmatrix}$	$\begin{pmatrix}0 & -1\\-1 & 0\end{pmatrix}$
설명	identity	y축반사	x축반사	180˚ 회전	y=x반사	90˚ 회전	270˚ 회전	y=-x반사

다. 만약 유사성이 존재하지 않는 레인지 블록들에 같은 도메인 블록을 대응시키면 식(1)과 (2)를 만족하는 IFS를 발견하지 못할 가능성이 증가하게 된다. 따라서 레인지 블록을 유사한 블록들의 그룹으로 분류하는 작업이 필요하다.

3.1 레인지 블록간의 유사성 척도

프랙탈 영상 압축에 사용되는 아핀 변환 함수는 변수 s와 o로 표현되는 레인지 블록R과 도메인 블록 D간의 1차원 변환 $R=s\times D+o$ 이다. 1차원 변환이 적용된 도메인 블록이 레인지 블록과 가장 작은 차이를 보일 때 복원의 속도가 빨라질 뿐만 아니라 보다 정확하게 원래의 영상을 복원하게 된다. 따라서, 두 레인지 블록간의 유사성도 1차원 변환이 적용되었을 때의 차이를 이용하여야 한다.
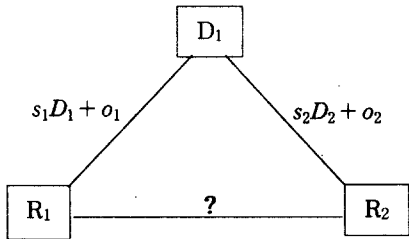
그림 1. 레인지 블록과 도메인 블록의 변환 관계

그림1과 같이 두 레인지 블록 R_1 과 R_2 는 같은 도메인 블록 D_1을 사용하여 원래의 영상을 빠르게 복원하기 위해서는 R_1과 D_1 사이의 변환 식 $s_1D_1+o_1$ 과 R_1과 R_2 사이의 변환 식의 결합이 R_2와 D_1 사이의 변환식 $s_2D_2+o_2$와 같아야 한다. 따라서 R_1과 R_2 사이의 변환 식은 반드시 1차 변환 식만 허용된다. 그리하여 두 레인지 블록간의 유사성을 판단하는 척도는 R_1과 R_2 사이의 1차 변환 내에서 두 블록이 얼마나 유사한 가를 측정하는 방법이 사용되어야 한다. 이때 R_1과 R_2 사이에 적용될 1차 변환은 지금까지 프랙탈 압축에서 사용되었던 두 영상간의 변환 방법과 같은 방법을 적용하는 것이 바람직하다. 즉, 레인지 블록과 도메인 블록간의 유사도를 구하던 식(1)의 변환을 레인지 블록간의 변환 식에 사용한다.

일반적으로 1차 변환된 두 영상간의 유사도는 두 영상간의 차이의 정도를 나타내는 척도로 두 영상을 구성하는 밝기 차의 제곱평균(MSE), 두 블록간의 최대 픽셀 밝기 값의 차 등 다양한 척도가 존재한다. 그러나 일반적으로 두 판단 척도인 픽셀 밝기 값의 차의 제곱평균인 MSE 값과 두 블록간의 최대 픽셀 밝기 값의 차가 널리 사용되고 있다. 이 이외에도 영상간의 변환을 사용하여 두 영상간의 유사성을 측정하는 방법도 사용할 수 있으나 MSE나 최대 픽셀 밝기 차만큼 직접적으로 두 영상의 차이를 잘 표현하지 못하고 있다. 이에 따라 본 연구에서는 두 영상간의 유사도에 대한 척도로 MSE와 최대 픽셀 밝기 차를 활용한 레인지 블록간의 유사도를 분석하게 될 것이다.

두 판단 척도 중 레인지 블록간의 MSE값에 대한 척도를 먼저 살펴보기로 하자. 두 레인지 블록 R 과 T를 구성하는 픽셀을 R_1 .... R_n 그리고 T_1 .... T_n라 할 때 두 블록간의 픽셀 밝기 값의 차이를 이용한 MSE값은 식(5)와 같이 구해진다.

$$MSE(R,T) = \frac{\sum_{k=1}^{n}(R_k - T_k)^2}{n} \qquad (5)$$

여기서 n은 레인지 블록의 크기가 8×8 영상이므로 64가 되고, 레인지 블록 T에는 1차 식으로 표1의 등장 변환이 적용된 모든 결과 영상을 고려하게 된다.

두 번째 판단 척도인 최대 픽셀 밝기 값의 차이는 두 레인지 블록 R과 T를 구성하는 각 픽셀 밝기 값의 차이 중 최대 차이를 보이는 값 $\max(|R_i - T_i|)$을 의미한다. 두 레인지 블록 R 과 T 가 많은 부분에서 거의 유사한 값을 갖는데 몇몇의 픽셀에서만 특이하게 높은 밝기 차이를 보이는 경우 두 영역을 유사하다고 판단하면 전체 결과에서 해당되는 부분은 복원 시 소멸 될 것이다. 그러나 두 블록을 유사하지 않다고 판단하게 되면 해당 부분을 원래의 모양대로 복원하는 것이 가능해진다. 즉, 하나의 예로 탁구 영상에서 공은 흰색을 띄고 있다. 이 공의 크기가 매우 작을 경우 평균값만 활용을 한다면 공을 포함한 레인지 블록이 다른 레인지 블록과 유사하다고 판단되어 공의 영상을 잃어버리는 우를 범할 가능성이 많이 존재하게 된다.

3.2 유사도에 따른 레인지 블록의 분류

본 연구에서는 모든 레인지 블록들을 유사 레인지 블록들의 집합(그룹)으로 나누기 위해 각 레인지 블록들간에 유사도에 대한 측정이 이루어져야 한다. 즉, A, B, C, D의 레인지 블록에 대한 유사성을 측정

한 결과 A 레인지 블록은 B, D와 유사한 블록으로 결정되었고, B는 A 와 C, C는 B 와 D, 그리고 D는 A 레인지 블록과 유사한 것으로 결정되었다고 하자. 이에 따라 4개의 레인지 블록은 유사한 블록끼리 그룹화가 되어져야 한다. 하나의 예로 A 레인지 블록을 B와 D 블록과 묶어 하나의 유사 레인지 블록 그룹을 만들고 C를 단독의 유사 레인지 블록 그룹으로 정의하자. 이때 첫 번째 유사그룹인 A, B, D블록에서는 B블록과 D 블록 사이에는 유사성이 없으나, B와 D 블록이 A 레인지 블록과 유사한 관계를 가지고 있다. 따라서 A 블록에 대응되는 도메인 블록을 찾아 B와 D 블록과의 변환 관계식을 도출해 낼 수 있게 된다. 즉, 하나의 유사 레인지 블록 그룹에 속하는 모든 레인지 블록간의 유사성이 아니라, 그룹에 속한 하나의 레인지 블록과의 유사성만 존재하면 프랙탈 압축 이론에 근거한 아핀 변환 식을 도출해 낼 수 있게된다. 그리하여 블록의 그룹화에서는 임의의 레인지 블록에 대해 특정 그룹에 속할 가능성을 계산할 때 그룹내의 모든 레인지 블록이 아니라 그룹을 대표하는 하나의 레인지 블록과만 비교하면 된다.

레인지 블록의 그룹화를 위해 Greedy(욕심쟁이) 알고리즘 기법을 활용하여 레인지 영역을 분류하게 된다. 즉, 임의의 레인지 블록에 대해 현재까지의 레인지 블록의 분류 결과에 따른 각 그룹의 대표 블록과 비교하여 적절한 그룹에 소속시키는 방법을 사용하게 된다. 욕심쟁이 기법을 사용하기 위해 초기 상태는 어떤 레인지 블록도 유사도에 따른 분류가 이루어지지 않은 상태로 시작하게 된다. 각 유사 레인지 블록 그룹에서의 대표 블록은 계산의 편리를 위해 해당 그룹이 처음 만들어질 때 고려된 레인지 블록으로 가정한다. 임의의 레인지 블록 r_i가 속할 유사 그룹을 찾기 위해서는 현재까지 고려된 레인지 블록들($r_0 \sim r_{i-1}$)의 분류 결과 발생한 각 유사 그룹의 대표 블록들과 유사성을 계산한다. 즉, 현재 i개의 레인지 블록이 k개의 레인지 블록 그룹($G_1 \sim G_k$)으로 분류되고 각 그룹의 대표 블록을 A_m라 할 때 임의의 레인지 블록 r_i는 A_m과 비교가 이루어진다. 즉, 일반적인 프랙탈 압축 기법에 사용되었던 8개의 아핀 변환I를 r_i에 적용시킨 변환 블록과 A_m과의 MSE 값을 계산해서 MSE 값이 가장 작은 대표 블록 A_{min}을 찾는다.

$$A_{\min} = \min_{1 \le m \le k} [MSE(A_m, I(r_i))] \quad (6)$$

최소 MSE 값이 미리 정의된 문턱치(Threshold)보다 작으면 레인지 블록 r_i는 대표 레인지 블록 A_{min}이 속한 그룹에 포함된다. 이때 적용된 아핀 변환의 계수인 레인지 블록 r_i와 A_{min}의 변환 규칙은 3.3절에서 설명될 r_i 블록의 프랙탈 압축 결과의 유추에 사용된다. 만일 최소 MSE 값이 문턱치보다 클 경우에는 r_i가 속할 유사 그룹이 없기 때문에 새로운 유사 그룹 G_{k+1}을 생성하고 레인지 블록 r_i를 유사 그룹 G_{k+1}의 대표 블록 A_{k+1}로 정의한다. 이 과정을 모든 레인지 블록에 차례로 적용시킴으로써 전체 레인지 블록을 유사 레인지 블록들의 그룹으로 분류를 수행하게 된다.

3.3 대표 블록의 변환 값에서 각 레인지 블록의 변환 값 계산

입력 영상에 대한 레인지 블록들을 유사도에 따라 레인지 블록의 그룹으로 분류한 후 각 그룹의 대표 블록에 대해 도메인 탐색을 수행하여 프랙탈 압축을 시도하게 된다. 각 그룹의 대표 레인지 블록에 대한 도메인 탐색의 결과 획득된 도메인 블록은 해당 대표 레인지 블록이 속한 그룹의 모든 레인지 블록에 적용되어 프랙탈 압축 계수를 구하게 된다. 즉, 그림 2와 같이 하나의 레인지 블록 그룹에 속하는 대표 레인지 블록(A)과 임의의 레인지 블록(r_i)간의 변환 규칙 I와 대표 레인지 블록과 도메인 블록(D_{mse})간의 변환 규칙 G를 이용해서 임의의 레인지 블록 r_i와 도메인 블록 D_{mse}간의 변환 규칙을 구하게된다.

그림 2. 레인지 블록과 대표 레인지 블록의 도메인 블록간의 관계

임의의 레인지 블록에 대응되는 도메인 블록은 해당 대표 레인지 블록의 탐색 과정에서 획득한 것과 같으므로 도메인 블록의 위치는 같다. 그리고 레인지 블록의 프랙탈 압축에 사용되는 등장변환은 그림 2에 표시된 바와 같이 유사 그룹의 분류 과정에서 기록해 둔 레인지 블록과 대표 레인지 블록간의 변환 규칙 I와 대표 블록과 도메인 블록간의 변환규칙 G의

곱으로 구할 수 있다. 이 행렬 곱셈의 결과가 표 2에 기술되어 있다. 그리고 도메인 블록D_{mse}과 레인지 블록r_i간의 평균 밝기 차를 나타내는 offset계수 o는 식 (7)과 같이 ri과 A블록간의 offset과 A와 D_{mse}블록간의 offset을 더함으로써 구해진다.

$$offset(r_i, D_{mse}) = offset(r_i, A) + offset(A, D_{mse}) \quad (7)$$

그리고 일반적으로 프랙탈 압축에서 두 영상간의 contrast scaling 계수인 s는 고정시켜 사용하기 때문에 r_i와 D_{mse} 블록간의 s값도 고정된 값을 그대로 사용하게 될 것이다. 만약 고정시키지 않았다면 그림1에서 보여주는 바와 같이 두 변환에서 구해진 각각의 s값을 곱한 결과를 사용하게 된다.

이와 같이 본 논문에서 제안한 방법은 유사 레인지 블록 그룹의 대표 블록을 제외한 나머지 레인지 블록에 대해서는 도메인 블록의 비교 탐색을 하지 않고 간단한 계산만으로 프랙탈 코드를 생성할 수 있으므로 그만큼의 부호화 시간을 줄일 수 있다. 표 2에서 R은 레인지 블록 r_i 와 도메인 블록D_{mse} 간의 기하 변환 규칙으로 최종적으로 각 레인지 블록의 압축에 사용되는 변환 규칙이 된다.

3.4 레인지 유사성을 고려한 경우의 도메인 탐색 횟수의 분석

입력 영상이 N×N 픽셀 크기이고, 레인지 블록이 $m \times m$, 도메인 블록이 $2m \times 2m$ 크기라고 하면 전체 레인지 블록 수는 $(N/m)^2$, 그리고 도메인 블록의 수는 $(N-2m+1)^2$이 된다. 탐색 영역의 제한 없이 전 탐색을 실시해 레인지 블록에 대한 프랙탈 코드를 생성한다면 하나의 레인지 블록에 대한 프랙탈 코드 생성을 위해 필요한 도메인 블록과의 비교 횟수는 전체 도메인 블록 수인 $(N-2m+1)^2$이 된다. 유사 그룹의 분류를 위한 레인지 블록간의 비교 횟수는 분류되는 유사 그룹의 수에 따라 다르겠지만 최악의 경우 같은 그룹에 속하는 레인지 블록이 하나도 없다고 가정했을 때 $0+1+2+\ldots+(N^2/m^2-1) = (N^4-m^2N^2)/2m^4$이 된다. 그래서 유사 그룹의 분류에 필요한 비교 횟수와 도메인 영역 탐색에 대한 비교 횟수에 대한 비는 약 $(N^3-m^2N)/(2m^4N-8m^5)$이다. 즉, 같은 그룹에 속하는 레인지 블록의 수가 $(N^3-m^2N)/(2m^4N-8m^5)$만 되면 유사 그룹 분류로 인한 오버헤드가 없음을 알 수 있으며, 전체 레인지 블록 수에 대한 비율은 $(N-m^2)/(2m^2(N-4m))$이다.

일반적으로 가장 많이 사용하는 예제인 입력 영상의 크기가 256×256이고 m이 8인 경우 총 레인지 블록 수는 1024개, 도메인 블록 수는 58081개, 그리고 총 도메인 탐색 비교 횟수는 표1의 등장변환을 고려한 약 475백만($\cong$ 1024×58081×8)번이 된다. 그러나 본 연구에서 제안한 방법대로 총 1024개의 레인지 블록이 약 1000개의 유사 그룹으로 분류된다고 가정하면 레인지 블록의 분류에 최대 (1+2+⋯+1000+1000*24)*8 $\cong$ 4,200,000번의 레인지 블록간의 비교가 이루어지고, 대표 레인지 블록 1000개와 도메인 블록간의 비교횟수는 1000×58081×8$\cong$465백만이 되어 총 469백만 정도의 영상 비교가 이루어진다. 따라서 $(N^3-m^2N)/(2m^4N-8m^5) \cong 10$ 보다 큰 24개의 레인지 블록 유사성으로 영상 압축에 필요한 블록간의 총 비교 횟수가 줄어들었음을 알 수 있었다.

표 2. 레인지 블록과 도메인 블록간의 변환행렬 매핑

I	G	R	I	G	R	I	G	R	I	G	R	I	G	R	I	G	R	I	G	R	I	G	R
0	0	0	1	0	1	2	0	2	3	0	3	4	0	4	5	0	5	6	0	6	7	0	7
0	1	1	1	1	0	2	1	3	3	1	2	4	1	5	5	1	4	6	1	7	7	1	6
0	2	2	1	2	3	2	2	0	3	2	1	4	2	6	5	2	7	6	2	4	7	2	5
0	3	3	1	3	2	2	3	1	3	3	0	4	3	7	5	3	6	6	3	5	7	3	4
0	4	4	1	4	6	2	4	5	3	4	7	4	4	0	5	4	2	6	4	1	7	4	3
0	5	5	1	5	7	2	5	4	3	5	6	4	5	1	5	5	3	6	5	0	7	5	2
0	6	6	1	6	4	2	6	7	3	6	5	4	6	2	5	6	0	6	6	3	7	6	1
0	7	7	1	7	5	2	7	6	3	7	4	4	7	3	5	7	1	6	7	2	7	7	0

4. 실험 및 결과

본 논문에서 제안한 레인지 블록들간의 유사성을 분석하기 위해 프랙탈 압축에 자주 사용하는 gray scale 영상 4개를 선택하였다. 이 실험 영상으로는 그림 3에서 보여주는 baboon, lena, girl, bird 영상이다. Lena 영상과 girl 영상은 레인지 블록내의 밝기 변화가 적절히 안배되어 있는 영상으로 선택되었고, bird 영상은 배경 영상이 거의 일정하여 유사 레인지 블록이 많이 발견되는 예제로 선택하게 되었다. 그리고 baboon 영상은 레인지 블록 내의 밝기 변화가 극심한 경우로 레인지 블록의 그룹화가 힘든 예제로써 선택되었다.

위 네 영상에 대해 레인지 영역의 유사성을 측정하기 위해 본 연구에서는 MSE 값을 사용하여 유사도에 대한 기준치를 정하였다. 레인지 블록의 크기를 8×8로 한 경우 MSE에 따른 레인지 블록 그룹 수를 표시한 그림 4의 결과에서 baboon 영상은 레인지 블록간의 유사성이 상대적으로 매우 적다. 즉, 그림 4에서 MSE 값의 문턱치를 약 30으로 할 때 많은 레인지 블록이 그룹화가 이루어지기 때문에 본 연구가 효율적임을 알 수 있다. 그러나 baboon 영상에서 문턱치로 약 10 이하의 값들을 사용할 경우 레인지 블록의 사전 실행이 비효율적인 것으로 나타났다.

그림 5는 레인지 블록의 크기가 4×4인 경우에 대한 실험으로 8×8 크기의 레인지 블록의 경우보다 레인지 블록간의 유사성이 훨씬 많이 발견되었다. 즉, 4×4의 크기로 레인지 블록을 구분할 때 각 레인지 블록 내부의 밝기 변화에 대한 경우의 수가 적기 때문에 큰 레인지 블록의 경우보다 레인지 블록의 유사성이 매우 높게 나타나게 되었다. 이 두 실험 결과를 바탕으로 우리는 레인지 블록의 유사성을 판단

(a) baboon

(b) lena

(c) girl

(d) bird

그림 3. 실험 예제 영상
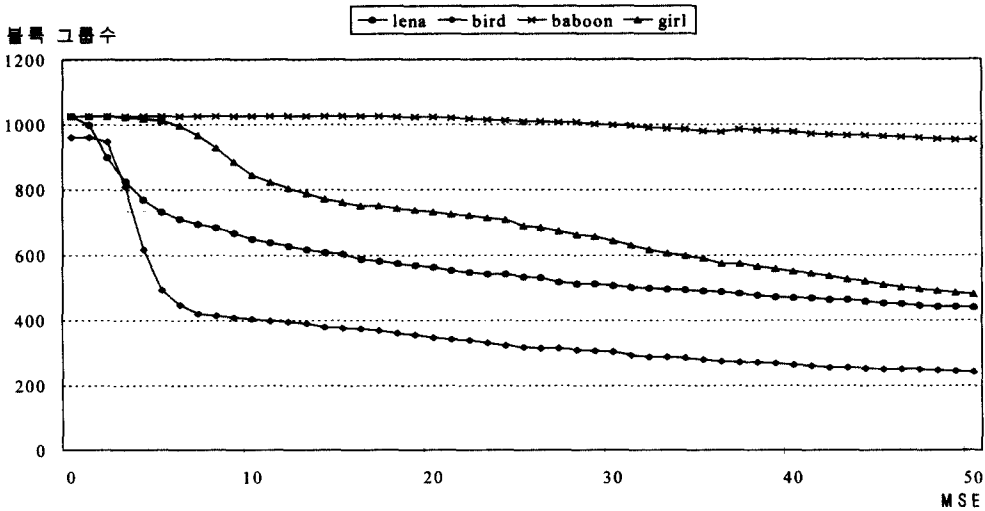
그림 4. 8×8 레인지 블록에서의 MSE 값과 유사 블록 그룹 수
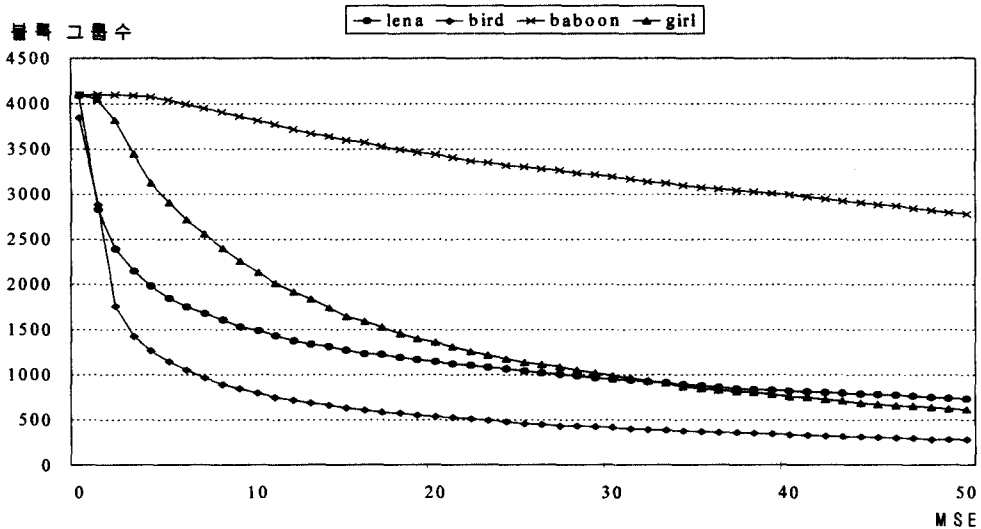
그림 5. 4×4 레인지 블록에서의 MSE값과 유사 블록 그룹 수

하기 위해 MSE 값 10을 문턱치로 압축을 수행하면 도메인 블록의 탐색을 수행하는 레인지 블록의 수가 대부분 1/2 이하로 줄기 때문에 전체적인 압축시간을 상당량 줄일 수 있을 것으로 기대 된다.

두 번째 유사도 측정 방법인 픽셀의 최대 밝기 차를 활용한 유사블록의 분류 결과가 그림 6에 표시되었다. 8×8 레인지 블록을 활용한 그림 6을 관측하면 배경 영상이 거의 비슷한 bird 영상의 경우는 최대 밝기 차가 5인 경우 약 50% 이상의 블록이 유사 그룹으로 분류된다. Lena 영상의 경우에는 최대 밝기차가 10 정도인 경우 전체 레인지 블록 중 30 % 이상이 유사 블록으로 분류된다. 그러나 baboons 영상은 MSE 값을 활용한 경우와 비슷하게 최대 밝기 차가 10 이상이 되어야 유사 블록이 발생함을 알 수 있다. 그러므로 일반적으로 10 정도의 밝기 차를 보이는 경우에는 대부분의 영상이 20% ~ 70% 정도가 유사 블록으로 분류되나 baboons 영상과 같이 이웃 픽셀 간의 밝기 변화가 심한 영상에서는 유사 블록이 거의
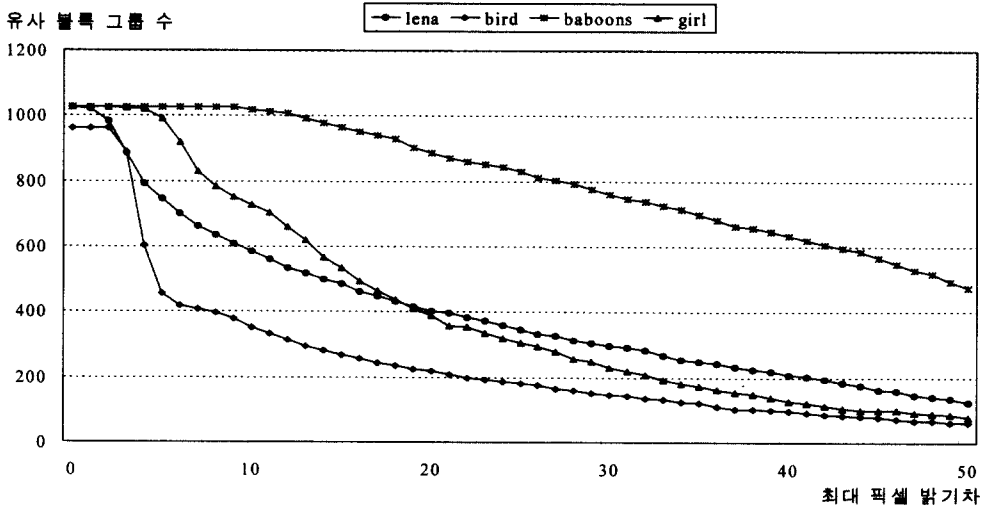
그림 6. 8×8 레인지 블록에서의 최대 밝기차를 이용한 유사 블록 수
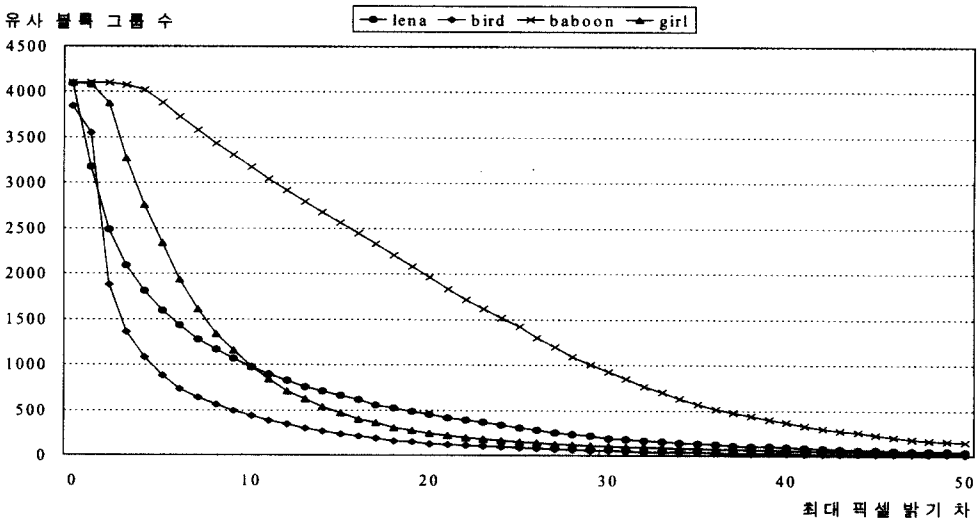
그림 7. 4×4 레인지 블록에서의 최대 밝기차를 이용한 유사 블록 수

없음을 알 수 있었다. 그림 7은 4×4의 레인지 블록을 사용한 경우의 유사블록 그룹 수를 보여주고 있다. 따라서 픽셀의 최대 밝기 차를 사용할 때는 약 10 이하의 문턱치를 사용하는 것이 바람직하다.

본 연구에서 제안한 레인지 블록의 유사성을 활용한 알고리즘의 개발 가능성을 보여주기 위해 그림 3의 실험예제 중 레인지 블록간의 유사성이 잘 드러나는 Lena와 bird 영상에 대해 압축/복원을 수행한 실험 결과가 표3에 제시되어 있다. 이 실험은 레인지 영역의 유사성이 잘 드러나는 4×4의 레인지 블록을 활용하여 프랙탈 압축의 대표적인 방법인 Barnsley가 제시한 기법에 적용하여 영상의 압축과 복원을 실행하였다. 표3에서 문턱치는 MSE 값과 유사한 SSE 값을 사용하였다. 표3에서 총 레인지 블록의 수 4096개가 문턱치가 커짐에 따라 많은 수의 블록이 유사 그룹으로 판명되었다. 이 유사 블록 그룹수의 감소와 거의 같은 비율로 전체적인 수행 시간도 단축되었다.

영상의 압축・복원 후의 품질을 나타내는 PSNR

표 3. 제안 방법의 효과

		Barnsley	문턱치(Threshold)			
			50	100	150	200
Lena	유사그룹수		2125	1730	1507	1351
	PSNR[dB]	33.31	33.19	33.10	32.99	32.92
bird	유사그룹수		1395	1023	823	702
	PSNR[dB]	36.91	36.68	36.37	36.25	36.11

값은 표3에서 보듯이 문턱치의 크기에 비례하여 감소하지만 Barnsley가 제안한 방법과 큰 차이를 보이지 않고 있다. 즉, 레인지 유사성을 적용한 경우 전체적인 영상의 품질은 아주 작은 정도만 저하됨을 알 수 있었다. 그리하여 본 제안 방법은 약간의 영상 품질의 저하로 획기적인 압축 시간의 단축을 가져오는 좋은 방법이 될 것이다.

5. 결 론

본 연구에서는 프랙탈 압축 방법의 단점인 긴 압축 시간을 줄이기 위해 압축의 사전 작업으로 사용될 수 있는 레인지 블록간의 유사성에 대한 분석 및 실험을 수행하였다. 레인지 블록의 유사성을 측정하기 위해 MSE 값과 최대 픽셀 밝기 차를 활용하였다.

레인지 블록을 그룹화 할 때 3.4절에서 증명한 바와 같이 8×8의 레인지 블록일 경우 약 0.6% 이상의 레인지 블록이 유사 레인지 블록으로 분류된다면 총 압축 시간을 줄일 수 있는 것으로 증명되었다. 또한 실험 결과인 그림 4와 그림 5를 통해 MSE값이 10 이하인 경우에 많은 영상이 20% 이상의 레인지 블록이 유사블록으로 분류되는 것을 알 수 있었다. 따라서 영상에서 이웃 픽셀들의 밝기 차 변화가 큰 baboon과 같은 영상에서는 본 방법이 효율적이지 못하였다.

픽셀의 최대 밝기 차를 활용한 방법에서는 밝기 차가 10이하가 되면 약 20% ~ 70% 정도가 유사 블록으로 분류됨을 알 수 있었다. 그러나 이 측정 단위를 사용한 방법은 MSE를 사용한 방법과 유사한 결과를 보여주었다. 따라서 두 측정단위 중 하나만 사용해도 충분한 것으로 판명되었다.

참 고 문 헌

[1] A. E. Jacquin. "Image coding based on a fractal theory of iterated contractive image transformations," *IEEE Transactions on Image Processing*, Vol. 1, No. 1, pp. 18-30, Jan. 1992.

[2] D. M. Monro and F. Dudbridge. "Fractal approximation of image blocks," *Proceedings ICASSP-92*, Vol. 3, pp. 485-488, 1992.

[3] Y. Fisher. *Fractal Image Compression*, SIGRAPH 92' Course Notes.

[4] M. F. Barnsley and L. P. Hurd. *Fractal Image Compression*. AK Peters, Wellesley, USA, 1993.

[5] K. U. Barthel, J. Schütemeyer, T. Voyé and P. Noll. "A new image coding technique unifying fractal and transform coding," *Proceedings ICIP-94*, Vol. 3, pp. 112-116, 1994.

[6] D. Saupe. "Fractal image compression via nearest neighbor search," In *NATO ASI on Fractal Image Encoding and Analysis*, Trondheim, Norway, Jul. 1995.

[7] B. Bani-Eqbal. "Speeding up fractal image compression," *Proceedings SPIE95*, Vol. 2418, pp. 67-74, 1995.

[8] B. E. Wohlberg and G. de Jager. "On the reduction of Fractal Image Compression encoding time," *COMSIG'94*, pp. 158-161, 1994.

[9] John Kominek. "Algorithm for Fast Fractal Image Compression," *Proceedings SPIE95*, Vol. 2419, pp. 296 -305, 1995.

[10] O. C. Au, M. L. Lion and L. K. Ma. "Fast Fractal Encoding in Frequency Domain," *IEEE Trans. on Image Processing*, Vol. 7, pp. 298-301, 1997.

김 영 봉

1987년 서울대학교 계산통계학과 (학사).
1989년 한국과학기술원 전산학과 (석사).
1994년 한국과학기술원 전산학과 (박사).
1994년 3월 ~ 1995년 2월 삼성전자 정보기술연구소 선임연구원.
1995년 3월 ~ 현재 부경대학교 컴퓨터멀티미디어공학부 조교수. 관심분야는 컴퓨터그래픽스, 애니메이션, 영상신호처리, 지리정보 시스템 임.